

What is claimed is:

1 1. A method for adapting a standard code base, the method comprising:
2 canonically parsing a modified version of a first release of a standard code
3 base to generate a canonically-parsed representation of the modified version;
4 generating difference data representative of changes made to the first
5 release of the standard code base using the canonically-parsed of the modified
6 version; and
7 using the difference data in applying the changes made to the first release
8 of the standard code base to a second release of the standard code base.

1 2. The method of claim 1, further comprising canonically parsing an unmodified
2 version of the first release of the standard code base to generate a canonically-parsed of
3 the unmodified version, wherein generating the difference data includes comparing the
4 canonically-parsed representations of the unmodified and modified versions of the first
5 release of the standard code base.

1 3. The method of claim 1, further comprising canonically parsing an intermediate
2 version of the first release of the standard code base to generate a canonically-parsed
3 representation of the intermediate version, wherein generating the difference data
4 includes comparing the canonically-parsed representations of the intermediate and
5 modified versions of the first release of the standard code base.

1 4. The method of claim 3, wherein the intermediate version of the first release of
2 the standard code base is generated using automated source transformation, and wherein
3 the modified version of the first release of the standard code base is generated by applying
4 manual changes to the intermediate version of the first release of the standard code base.

1 5. The method of claim 1, wherein generating the difference data includes
2 identifying a plurality of changed semantic components in the modified version of the
3 first release of the standard code base.

1 6. The method of claim 5, wherein identifying the plurality of changed semantic
2 components includes identifying a change made to a selected semantic component,
3 wherein the change is selected from the group consisting of deletion, modification,
4 addition and replacement.

1 7. The method of claim 6, wherein generating the difference data includes
2 generating at least one XML file, the XML file including a tag for a changed semantic
3 component, the tag identifying the changed semantic component and including an
4 attribute representing the change made to the changed semantic component.

1 8. The method of claim 5, wherein using the difference data in applying the
2 changes made to the first release of the standard code base to the second release of the
3 standard code base includes notifying a user of a change in a changed semantic
4 component.

1 9. The method of claim 5, wherein using the difference data in applying the
2 changes made to the first release of the standard code base to the second release of the
3 standard code base includes automatically applying a change in a changed semantic
4 component to the second release of the standard code base.

1 10. The method of claim 1, further comprising using the difference data in
2 applying the changes made to the first release of the standard code base to a third release
3 of the standard code base.

11. An apparatus, comprising:

a memory;

at least one processor; and

program code resident in the memory and configured to execute on the at least one processor to adapt a standard code base, the program code configured to canonically parse a modified version of a first release of a standard code base to generate a canonically-parsed representation of the modified version; generate difference data representative of changes made to the first release of the standard code base using the canonically-parsed of the modified version; and use the difference data in applying the changes made to the first release of the standard code base to a second release of the standard code base.

12. The apparatus of claim 11, wherein the program code is further configured to

canonically parse an unmodified version of the first release of the standard code base to

generate a canonically-parsed of the unmodified version, and wherein the program code is

configured to generate the difference data by comparing the canonically-parsed

representations of the unmodified and modified versions of the first release of the

standard code base.

13. The apparatus of claim 11, wherein the program code is further configured to

canonically parse an intermediate version of the first release of the standard code base to

generate a canonically-parsed representation of the intermediate version, and wherein the

program code is configured to generate the difference data by comparing the canonically-

parsed representations of the intermediate and modified versions of the first release of the

standard code base.

14. The apparatus of claim 13, wherein the intermediate version of the first

release of the standard code base is generated using automated source transformation, and

wherein the modified version of the first release of the standard code base is generated by

4 applying manual changes to the intermediate version of the first release of the standard
5 code base.

1 15. The apparatus of claim 11, wherein the program code is configured to
2 generate the difference data by identifying a plurality of changed semantic components in
3 the modified version of the first release of the standard code base.

1 16. The apparatus of claim 15, wherein the program code is configured to
2 identify the plurality of changed semantic components by identifying a change made to a
3 selected semantic component, wherein the change is selected from the group consisting of
4 deletion, modification, addition and replacement.

1 17. The apparatus of claim 16, wherein the program code is configured to
2 generate the difference data by generating at least one XML file, the XML file including a
3 tag for a changed semantic component, the tag identifying the changed semantic
4 component and including an attribute representing the change made to the changed
5 semantic component.

1 18. The apparatus of claim 15, wherein the program code is configured to use the
2 difference data in applying the changes made to the first release of the standard code base
3 to the second release of the standard code base by notifying a user of a change in a
4 changed semantic component.

1 19. The apparatus of claim 15, wherein the program code is configured to use the
2 difference data in applying the changes made to the first release of the standard code base
3 to the second release of the standard code base by automatically applying a change in a
4 changed semantic component to the second release of the standard code base.

- 1 20. The apparatus of claim 11, wherein the program code is further configured to
- 2 use the difference data in applying the changes made to the first release of the standard
- 3 code base to a third release of the standard code base.

1 21. A program product, comprising:

2 program code configured to adapt a standard code base by canonically
3 parsing a modified version of a first release of a standard code base to generate a
4 canonically-parsed representation of the modified version; generating difference
5 data representative of changes made to the first release of the standard code base
6 using the canonically-parsed of the modified version; and using the difference
7 data in applying the changes made to the first release of the standard code base to
8 a second release of the standard code base; and
9 a signal bearing medium bearing the program code.

1 22. The program product of claim 21, wherein the signal bearing medium
2 includes at least one of a transmission medium and a recordable medium.